

Comparing multiagent systems research in combinatorial auctions and voting

Vincent Conitzer

Published online: 24 July 2010
© Springer Science+Business Media B.V. 2010

Abstract In a combinatorial auction, a set of items is for sale, and agents can bid on subsets of these items. In a voting setting, the agents decide among a set of alternatives by having each agent rank all the alternatives. Many of the key research issues in these two domains are similar. The aim of this paper is to give a convenient side-by-side comparison that will clarify the relation between the domains, and serve as a guide to future research.

Keywords Multiagent systems · Combinatorial auctions · Computational social choice · Voting

Mathematics Subject Classifications (2010) 68-02 · 68T42 · 91B12 · 91B26

1 Introduction

In multiagent systems, it is often necessary for a group of agents to make a collective decision even though they have different preferences over the different options (a detailed discussion can be found in the article [31]). For example, the agents may have to decide how to allocate a set of items among themselves. A common mechanism for doing this is to run a *combinatorial auction*, where the agents place

This paper corresponds to an invited talk in the session on Computation and Social Choice at the Tenth International Symposium on Artificial Intelligence and Mathematics (ISAIM-08). An early version of this paper appeared in the informal electronic proceedings of ISAIM-08. The author is supported by an Alfred P. Sloan Research Fellowship and by NSF under award numbers IIS-0812113 and CAREER-0953756.

V. Conitzer (✉)
Department of Computer Science, Duke University, Durham, NC 27708, USA
e-mail: conitzer@cs.duke.edu

bids on bundles (subsets) of items (for example, an agent can bid 10 on the bundle consisting of items a and c), and based on these bids an allocation of the items is determined, as well as the payment that each agent needs to make. However, there are many other collective decision problems that do not involve allocating items or making payments. A general approach for choosing among a set of alternatives is for each agent to rank all the alternatives, after which a winning alternative is chosen based on these rankings. In this case, we say that the agents *vote* over the alternatives (and the rankings are the *votes*).

Combinatorial auctions have become a well-established research topic in multi-agent systems. In recent years, research on voting in multiagent systems has also soared, and a community of researchers interested in *computational social choice* has formed. While there is certainly significant interaction between the researchers working on combinatorial auctions and the ones working on voting, the two communities are perhaps more disjoint than one would expect. This paper aims to compare some of the research issues across the two topics, thereby providing to each community a convenient window into the other (and, perhaps, a window into both for outsiders). The comparison is also likely to suggest new research directions for each topic. Finally, we argue that the research on the two topics is likely to converge further in the future.

2 Defining the rules

Both in combinatorial auctions and in voting settings, we need to specify how agents report their preferences (that is, how they place their bids/cast their votes), and how an outcome is chosen on the basis of these. As we will see, there are multiple ways of doing so in each case.

2.1 Combinatorial auctions

To understand the different ways in which a combinatorial auction can be designed, it is helpful to first study some common single-item auctions.

- *English auction.* In an English auction (perhaps the best-known auction format) any bidder can enter a bid higher than the current highest bid at any point. Once nobody wants to submit a higher bid, the current highest bidder wins the item and pays her bid.
- *Japanese auction.* In a Japanese auction, there is an initial price of zero for the item, which is gradually increased. A bidder can leave the room at any point if the price becomes too high for her. The auction ends when only one bidder remains, who then wins and pays the final price.
- *Dutch auction.* In a Dutch auction, the price starts at a high value and is gradually *decreased*. At any point, any bidder can claim the item at the current price, at which point the auction ends.
- *First-price sealed-bid auction.* Each bidder privately sends a bid to the auctioneer (for example, in a sealed envelope). The highest bidder wins and pays her bid.
- *Second-price sealed-bid (aka. Vickrey) auction.* This auction is identical to the first-price sealed-bid auction, except the highest bidder (who still wins) pays the bid of the *second*-highest bidder.

A common objective among all of these auctions is to allocate the item to the bidder who values the item the most (that is, to allocate the item *efficiently*). The differences among them mostly reflect other issues, such as the following. One advantage of the English, Japanese, and Dutch auctions is that, depending on what happens in the auction, a bidder may not have to invest the effort necessary to determine her exact valuation for the item. For example, in an English or Japanese auction, once the second-to-last bidder drops out, the winner no longer has to think about how much longer she would have stayed in the auction; and in a Dutch auction, once someone else claims the item, the remaining bidders no longer have to think about when they would have claimed the item. This is closely related to *preference elicitation*, which we will discuss in a later section. An advantage of the second-price sealed-bid auction over the first-price sealed-bid auction is the following. In the first-price auction, bidders will try to only slightly outbid their competitors, to pay as little as possible. If the bidder with the highest valuation for the item underestimates the other bids, this can result in another bidder winning the item, so that the allocation is inefficient. In the second-price sealed-bid auction, however, there is no reason to try to only slightly outbid the next bidder; in fact, it can be shown that it is optimal to bid one's true valuation for the item. We will discuss issues such as these in a later section.

The main point, however, is that the various auctions agree, in some sense, on the objective of efficient allocation, and the differences merely reflect other issues. There are other auctions that have a different objective, such as revenue-maximizing auctions [83], which do not always allocate the item efficiently; or even auctions that try to minimize revenue (although most of these still allocate the item efficiently) [3, 10, 20, 63, 64, 80, 93]. Nevertheless, in combinatorial auctions, typically the main objective is to allocate efficiently (though there is some work on revenue-maximizing combinatorial auctions as well [5, 9, 38, 76, 77]). That is, it is assumed that every bidder i has a valuation function $v_i : 2^I \rightarrow \mathbb{R}$, where $v_i(S)$ is bidder i 's value for the subset S of the items I , and the goal is to allocate to the bidders nonoverlapping bundles S_i to maximize $\sum_i v_i(S_i)$. As in the single-item auction case, there are distinctions among combinatorial auctions in terms of the temporal aspects of the auction as well as the payments to be made by the bidders, but these distinctions are again driven by other considerations, which we will discuss in later sections.¹

2.2 Voting

In a typical voting setting, there is a set of *alternatives* or *candidates*, C , and each agent (voter) i has preferences \succeq_i over these m alternatives, with $a \succeq_i b$ indicating that i weakly prefers a over b —that is, i either strictly prefers a to b , or is indifferent

¹There are also variants of combinatorial auctions, such as combinatorial *reverse* auctions, where the auctioneer seeks to *buy* certain items and the bidders offer to *sell* bundles of these items at various prices; and combinatorial *exchanges*, where agents can be both buyers and sellers [105]. In another variant called *mixed multi-unit combinatorial auctions*, agents can also bid to transform one set of goods into another set of goods [21]. While these variants are very important, they resemble regular combinatorial (forward) auctions in terms of the issues discussed in this article, so we will not consider them in the remainder.

between the two. Usually, for convenience, it is assumed that all preferences are strict, so that we simply use \succ_i . (Recent work has begun to extend social choice theory to settings where there is *incompleteness/incomparability* in the agents' preferences [91, 92, 98].) Based on the preferences that the voters report, one alternative is chosen as the winner. (Sometimes, the output is a complete (aggregate) ranking of the alternatives; generally, a rule can be used to determine either just a winner or an aggregate ranking.) Some example rules are given below. For each rule that gives points to alternatives, the alternative with the highest score is the winner. Generally, some tie-breaking scheme is required.

- *Plurality*. An alternative receives a point every time it is ranked first.
- *Borda*. An alternative receives $m - 1$ points every time it is ranked first, $m - 2$ every time it is ranked second, ..., and 0 every time it is ranked last.
- *Copeland*. We conduct a *pairwise election* between every pair of alternatives: the winner of the pairwise election is the alternative that is ranked higher by more voters. An alternative receives 2 points for every pairwise win, 1 point for every pairwise tie, and 0 points for every pairwise loss.
- *Bucklin*. If there is an alternative that is ranked first by more than half the voters, that alternative wins; otherwise, if there is an alternative that is ranked first or second by more than half the voters, that alternative wins; etc.
- *STV*. The alternative that is ranked first the *fewest* times is removed from every vote. (Votes that had this alternative ranked first will now have another alternative ranked first.) This is repeated until one alternative remains.
- *Slater*. We choose an aggregate ranking of the alternatives that is consistent with the outcomes of as many pairwise elections as possible. (If a defeats b in their pairwise election, then ranking a ahead of b would be consistent with the outcome of this pairwise election, whereas ranking b ahead of a would not be.)
- *Kemeny*. We choose an aggregate ranking of the alternatives that has as few disagreements with votes as possible (a *disagreement* occurs when a vote ranks a above b , but the aggregate ranking ranks b above a ; the disagreements are summed over all votes and all pairs of alternatives).

In some sense, the differences among these rules are more fundamental than the differences among (combinatorial) auctions. Most auctions agree on the objective of efficient allocation, and the differences among them are due more to other considerations, such as incremental preference revelation and strategic bidding. In contrast, in a voting setting, it is not even clear what objective we should be pursuing. Certainly we can state a vague objective such as “the winner should be high in the preferences of many voters and low in the preferences of few,” but there are many different interpretations of this, and, in some sense, each of the voting rules above corresponds to one such interpretation.

One natural approach to identifying the “optimal” voting rule is to specify some axioms that such a rule should satisfy. Unfortunately, there are several impossibility results that show that *no* rule satisfies certain natural properties. For example, consider the *independence of irrelevant alternatives (IIA)* criterion, which states that, if we modify the votes but do not change whether a is ahead of b in any vote, then in the aggregate ranking, there also should be no change in whether a is ahead of b . While this is a natural criterion, *Arrow's impossibility theorem* [6] states that with 3 or more alternatives, any rule that satisfies IIA must either be *Pareto inefficient*, which

means that it sometimes ranks b ahead of a even though all voters rank a ahead of b , or *dictatorial*, which means that the rule simply copies a fixed voter's ranking. There are some combinations of other axioms that are satisfied by exactly one rule—for example, Young and Levenglick [131] give an axiomatic characterization of the Kemeny rule. However, such axiomatic arguments have so far not succeeded in building broad consensus on what the optimal rule is.

Another approach that has been pursued to decide on the optimal voting rule is the following. Imagine that there exists a “correct” outcome (winner or ranking), which we cannot directly observe; but every voter's preferences constitute a noisy observation of this correct outcome. Given a noise model (a conditional probability distribution over the preferences given the correct outcome), it would make sense to choose the outcome that maximizes the likelihood of the observed preferences. This approach was already pursued by the early social choice theorist Condorcet [46], who proposed one particular noise model. In this noise model, each voter ranks each pair of alternatives correctly with some constant probability $p > 1/2$, independently. (This can lead to cyclic preferences, but that does not affect the maximum likelihood approach.) Condorcet solved for the maximum likelihood estimator rule for the cases of two and three alternatives. Two centuries later, Young showed that the solution for general numbers of alternatives coincides with the Kemeny rule [130]. Unfortunately, this argument for the Kemeny rule is convincing only to the extent that one believes that Condorcet's noise model is the correct one. In fact, more recent results imply that for many (but not all) of the common voting rules, there is a noise model such that that rule becomes the solution [35, 39] (another paper [111] deals with the case of the Borda rule specifically). Another recent paper [51] explores the relationship between this maximum likelihood framework and the *distance rationalizability* framework, where we find the closest “consensus” election that has a clear winner, for some definition of consensus and some distance function, and choose its winner.

It thus appears that for the foreseeable future, no general agreement will emerge on which rule is optimal.² Moreover, as we move towards settings in which there are so many alternatives that it is no longer feasible for a voter to provide a full ranking of all of them, new rules must be designed to address this. We will discuss this in more detail in the next section.

3 Expressing preferences

In the previous section, we defined the structure of the agents' preferences (valuations for bundles in combinatorial auctions, and rankings of alternatives in voting). In order to choose an outcome based on the agents' preferences, the agents will need to report them; and to do so, the agents need a *language* in which to express their preferences. It is generally easy to create a straightforward language for doing so, but expressing one's preferences in such a language often requires exponential space.

²A notable exception is the case where there are only two alternatives: in this case, there are good reasons to consider the majority rule (the alternative preferred by more voters wins) optimal—and indeed common voting rules generally coincide with the majority rule in the two-alternative case.

A good language for expressing preferences will allow agents to specify “natural” preferences concisely.

3.1 Combinatorial auctions

In a combinatorial auction, the straightforward way for a bidder to communicate her preferences to the auctioneer is to simply provide a list of $2^{|I|} - 1$ values, one for each nonempty bundle of items. Of course, this is impractical for anything other than very small numbers of items. Hence, it becomes important to have a good *bidding language* in which bidders can express their valuation functions concisely. Unfortunately, regardless of the language used, there will always be *some* valuation functions that require exponential space to express, for purely information-theoretic reasons. However, a good bidding language allows bidders to express *natural* valuation functions concisely. This is analogous to, for example, Bayesian networks (for an introduction to Bayesian networks, see Chapter 14 of [101]): any distribution can be represented using a Bayesian network, but in general, doing so requires specifying an exponentially large number of probabilities. However, natural distributions display a significant amount of conditional independence, and such distributions can be expressed much more concisely using a Bayesian network.

So, the question becomes: which valuation functions are natural? One common assumption is that the bidder is *single-minded*, that is, there is a bundle S_i and a constant k_i such that $v_i(S) = k_i$ if $S_i \subseteq S$, and $v_i(S) = 0$ if $S_i \not\subseteq S$. That is, the bidder has her heart set on a particular bundle; if she gets it, she receives a utility of k_i (and any additional items that she receives will simply be thrown away), but if even one item from S_i is missing, the bundle becomes worthless to her. A single-minded valuation function is easy to express: in this case, a bid consists simply of a bundle S_i and a value k_i . However, usually, bidders are interested in more than one bundle. The *OR language*³ allows a bidder to bid on multiple bundles. An example bid in this language is $(\{a\}, 3) \text{ OR } (\{b, c\}, 4) \text{ OR } (\{c, d\}, 5)$. This bid indicates that if the bidder receives the bundle $\{a, b, c\}$, her value is 7 (because she receives the first two bundles in her bid); if she receives $\{b, c, d\}$, her utility is 5 (each of the last two bundles in her bid are contained in the bundle she receives, but c can only be counted towards one of them, and the last bundle has the greater valuation). A disadvantage of the OR language is that it cannot express every valuation function. For example, consider the valuation function $v(\{a\}) = 1$, $v(\{b\}) = 1$, $v(\{a, b\}) = 1$ (the bidder wants either item and has no use for a second item). If the bidder bids $(\{a\}, 1) \text{ OR } (\{b\}, 1)$, this implies a valuation of 2 for the bundle $\{a, b\}$; on the other hand, if she does not include both singleton bundles in her bid, then the bid will not reflect her valuation for those.

An alternative language is the *XOR language*. At most one of the bundles in an XOR-bid can be counted: for example, the bid $(\{a\}, 3) \text{ XOR } (\{b, c\}, 4) \text{ XOR } (\{c, d\}, 5)$ indicates that the bidder's valuation for $\{a, b, c\}$ is 4 (only one of the first two bundles can be counted). In this language, any valuation function can be expressed (if necessary, by XOR-ing all possible bundles together). One downside of the XOR language is that some natural valuation functions require exponential space to represent—for example, the function $v(S) = |S|$ (which can be concisely

³For a detailed discussion of the OR and XOR languages, see [85].

represented in the OR language by listing each singleton bundle at a value of 1). Of course, nothing prevents us from using ORs and XORs simultaneously, to obtain the best of both worlds.

There are also bidding languages of a different flavor. For example, the bidder can specify interactions among the items [23, 44]. If a bidder reports a valuation of 2 for item a , a valuation of 3 for item b , and an interaction of -1 between a and b , this implies a valuation of $2 + 3 - 1 = 4$ for the bundle $\{a, b\}$.

3.2 Voting

In a voting setting, a straightforward way for a voter to communicate her preferences is to simply communicate the position of each alternative in her ranking. If there are m alternatives (and hence m different positions), specifying one position requires $\Theta(\log m)$ bits, leading to a total space requirement of $\Theta(m \log m)$ bits. In many settings, this is quite manageable, and because of this the problem of how preferences are represented often does not receive much attention in voting.

Nevertheless, there are many important settings in which there are exponentially many alternatives, so that $\Theta(m \log m)$ space is no longer manageable. For example, in some settings, the set of alternatives may be written as $C = X_1 \times X_2 \times \dots \times X_p$. Here, each X_j corresponds to a separate *issue* on which we need to make a decision. In such a domain, we need to make use of a more sophisticated language for representing preferences, such as a CP-net [18]. (A CP-net allows an agent to specify whether and how her preferences on one issue depend on the values chosen for the other issues.) Preferences expressed in such a language do not always give enough information to recover the full ranking of all alternatives, so not all the standard rules can be applied in such a setting. Some very recent work has been devoted to determining how winners should be chosen in such settings [74, 121, 122, 124, 125], but much more remains to be done here.

4 Winner determination

Once we have defined the rules according to which an outcome is chosen, as well as the language in which preferences are reported, we have a well-defined computational problem of deciding what the outcome is given the reported preferences. In combinatorial auctions, this computational problem is usually called the *winner determination problem*. Since in voting, we are also determining a winner, we will use the same name for the problem in that setting.

4.1 Combinatorial auctions

The winner determination problem in a combinatorial auction is to determine the allocation that maximizes the total value, given the bids (expressed in some bidding language). For simplicity, let us first assume that every bidder i is single-minded, and hence her bid can be represented as (S_i, k_i) . The problem is then to select a subset A of the bids to accept, to maximize $\sum_{i \in A} k_i$, under the constraint that $S_i \cap S_j = \emptyset$ for all $i, j \in A$, $i \neq j$. This problem is NP-hard [100], even to approximate [102, 133]. There are various approaches to solving it nonetheless, including: modeling

it as an integer program, using a search-based algorithm [103], or using a dynamic programming algorithm [100].

It turns out that any algorithm for the winner determination problem with single-minded bids can be extended to deal with bids that use ORs and XORs. For ORs, this is easy to see: as far as winner determination is concerned, there is no difference between receiving two single-minded bids (S_1, k_1) and (S_2, k_2) , or receiving a single bid (S_1, k_1) OR (S_2, k_2) , because in either case the auctioneer can award both bundles for a value of $k_1 + k_2$ if and only if the bundles do not overlap. This argument does not work for XORs, but there is a clever trick for converting an XOR into an OR [58, 84]: given a bid (S_1, k_1) XOR (S_2, k_2) , create a dummy item d , and replace the bid by $(S_1 \cup \{d\}, k_1)$ OR $(S_2 \cup \{d\}, k_2)$. Now the two bundles cannot both be awarded to the bidder, because they overlap. Because of this, most research on the combinatorial auction winner determination problem has focused on single-minded bids.

While the winner determination problem with single-minded bids is NP-hard in general, it can become polynomial-time solvable if the bids lie in a restricted class [33, 61, 82, 90, 100, 104, 110]. For example, it is polynomial-time solvable if each bid is on at most two items [100]. It is also polynomial-time solvable if, in addition to the bids, we are given a graph of bounded treewidth whose vertices are the items and whose edges are such that every bid is on a connected set of vertices/items [33] (this result can be generalized further [61]).

For other bidding languages, the complexity of the winner determination problem may be different. For example, for the language based on specifying interactions among items (described above), the winner determination problem is NP-hard even with only pairwise interactions [23, 44].

4.2 Voting

In settings where the number of alternatives m is not extremely large, so that each voter can communicate her complete ranking of all the alternatives, determining the winning alternative is computationally straightforward for most voting rules. For example, a rule such as Borda requires nothing more than adding up the scores of the alternatives and comparing them. However, this is not the case for all voting rules: some of them are in fact NP-hard to execute [1, 2, 13, 19, 26, 27, 50, 67, 96, 99]. As an example, let us take the Slater rule, which requires finding a ranking that is inconsistent with the outcomes of as few pairwise elections as possible. Determining the outcomes of the pairwise elections is easy. It is helpful to summarize this information in a directed graph, in which the alternatives are the vertices and each edge points from the winner of a pairwise election to the loser. For simplicity, let us assume that none of the pairwise elections end up tied (for example, we can assume that the number of voters is odd), so that there is an edge between every pair of alternatives. (For every graph of this form, there exists a set of votes that results in this graph [78].) If this graph is acyclic, then it already corresponds to a ranking. If not, then the Slater rule asks us to make the graph acyclic, by reversing as few edges as possible. This problem turns out to be NP-hard [1, 2, 22, 27]. The Kemeny rule can similarly be interpreted in a graph-theoretic way: to do so, we use the same graph as for the Slater rule, except we add a weight to each edge (a, b) which is equal to the margin of a 's pairwise victory over b . The problem now becomes to reverse a set of edges of minimum total *weight* to make the graph acyclic. It is possible to find votes

so that all the weights become equal to each other [78], so the Kemeny problem is at least as hard as the Slater problem, that is, NP-hard. Still, it is possible to solve reasonably-sized winner determination instances for the Kemeny rule, using search-based techniques [45, 79] or integer programming [32]. Fixed-parameter tractability results for computing Kemeny rankings have also been given [15]. The remaining rules introduced earlier in this article (the ones other than Slater and Kemeny) can be executed in polynomial time.

5 Preference elicitation

So far, we have assumed that each agent reports her complete preferences all at once. While a good language for describing preferences can be very helpful in doing so, the agent still needs to determine her complete preferences before she can report them. An alternative approach is to use a sequential process, where one agent reports some information about her preferences; then, based on that information, another agent reports some information about his preferences, etc., until we know enough to determine the outcome. This approach is known as *preference elicitation*, and it is usually guided by a central party, the *elicitor*, who determines who has to report which information next by asking that agent a *query* about her preferences.

When we use preference elicitation, it is generally not necessary for each agent to reveal all of her preferences. Very often, once we have obtained some preference information from the agents, other information becomes irrelevant. For example, if we have already determined that one agent's valuation for a particular bundle is not high enough to have any chance of winning, then there may not be any need to determine the exact valuation. As another example, if we have already determined that one alternative has no chance of winning, then there may not be any need to determine its exact position in a voter's preferences. Making use of such observations can reduce the amount of information that the agents have to communicate. However, the benefit here is more than mere communicational convenience. Often, the greater benefit is that the agent does not even need to *determine* her preferences completely. A significant amount of deliberation effort can be required to determine even a single bit of preference information (for example, which of two alternatives is preferred), so it can be very valuable to realize that this information is not needed. An additional benefit is that the agents' privacy is improved, in the sense that they release less of their preference information. It follows that preference elicitation can be extremely valuable even in settings where preferences can be described concisely—for example, voting over a small set of alternatives.

5.1 Combinatorial auctions

In combinatorial auctions, it is easy to see the potential benefits of preference elicitation. Even when armed with the best bidding language, determining one's exact valuation for every bundle can be overwhelming. It is therefore not surprising that much of the work on preference elicitation has focused on combinatorial auctions.

In a sense, single-item auctions such as the English, Japanese, and Dutch auctions are already doing some preference elicitation. At each stage, bidders reveal some

information, such as whether the current price is too high for them; and once we know the winning bidder, we end the auction, even if we do not yet know all bidders' exact valuations. Indeed, some of the approaches to preference elicitation in combinatorial auctions mimic these single-item auctions. Several *ascending* combinatorial auctions have been proposed [7, 88, 89]. In every round of an ascending combinatorial auction, each bidder faces a price for each bundle, and needs to decide which bundle she would like to buy under these prices. The prices start low, so that bidders will choose large bundles that overlap with each other. Each round, prices are raised, up to the point that the chosen bundles no longer overlap with each other. While ascending combinatorial auctions can, under certain conditions, result in the optimal allocation, they also face some inherent limitations [17].

Ascending combinatorial auctions typically proceed in a very systematic fashion. It is also possible to take a more flexible preference elicitation approach, where the auctioneer/elicitor can ask any agent any query at any point. Common queries include the *demand query*, where a bidder is asked which bundle she would prefer under certain prices (as in the ascending auctions), and the *value query*, which simply asks for the bidder's valuation for a specific bundle.

One question that is often studied in preference elicitation is the following: if a bidder's valuation function is guaranteed to lie in a certain class of functions C , can we determine the bidder's valuation function exactly using only a polynomial number of queries? Various positive results have been obtained along this line. For example, a bidder's valuation function can be elicited exactly with a number of value and demand queries that is polynomial in the length of the valuation function's representation in the XOR language [73]. Hence, valuations that admit a concise XOR-representation can be elicited efficiently using such queries. However, this result only holds if the demand queries can set prices on bundles (rather than just on individual items) [16]. There are numerous other results that show that various classes can(not) be elicited using a polynomial number of queries of a given type [17, 44, 72, 107, 132].

If there are no restrictions on the valuation functions, some negative results are known. For example, it has been shown using techniques from communication complexity theory [71] that the winner determination problem in general requires exponential communication [87]. This is true regardless of the types of query that are allowed.

5.2 Voting

When agents are voting over a set of alternatives that is not extremely large, the potential benefits of preference elicitation are less dramatic than in a combinatorial auction: even reporting one's complete preferences requires only a polynomial amount of communication. Nevertheless, as we have already noted, determining these preferences generally still requires a large amount of deliberation effort. Hence, there is still significant value in doing preference elicitation in such settings.

As in the case of combinatorial auctions, there has been some work on determining how many queries are needed to completely elicit a voter's preferences, assuming that the preferences lie in a specific class. For instance, sometimes there is a natural order $<$ on the alternatives—for example, some candidates may be more “right-wing” than others. A voter's preferences $>_i$ are said to be *single-peaked* with respect

to this order if, whenever $a < b < c$ or $c < b < a$ (where a is i 's most-preferred alternative), we have that $b \succ_i c$. That is, the voter prefers alternatives that are closer to her most-preferred alternative. It has been shown that single-peaked preferences can be elicited using a linear number of *comparison queries* (which ask the voter which of two alternatives is preferred), if either the order $<$ is known, or at least one other vote that is single-peaked with respect to $<$ is known [30].

Another important topic is the communication complexity of executing various rules—that is, for a given rule, what is the minimum number of bits that needs to be communicated by the voters (in the worst case) to determine the winner?⁴ It should be noted that here, there is no constraint on the types of query that are used. Some of the rules, such as Borda and Copeland, turn out to require $\Omega(nm \log m)$ communication in the worst case—that is, in the worst case, the communication requirements are roughly as bad as having everyone reveal all of their preferences. For other rules, however, we can get away with much less communication: for example, STV requires only $O(n(\log m)^2)$ (using the straightforward protocol where everyone reports their most-preferred alternative first; then, when an alternative is eliminated, all voters who ranked that alternative first communicate their next-most preferred alternative, *etc.*), and Bucklin requires only $O(nm)$ (using a more sophisticated protocol based on binary search) [40].

Another key computational problem for preference elicitation is the following [36, 70, 92, 114, 117]: given some of the preferences of some of the voters, do we already know which alternative is the winner? If we can solve this problem, then we will know when we can stop eliciting preferences and declare the winner. This problem is often easy, but sometimes it is NP-hard, for example for STV [36].

A final direction is to optimize the elicitation process: given a prior distribution over the voters' preferences, plan the elicitation process to minimize the expected number of voters whose preferences we elicit before we know the winner. For most rules, this problem is NP-hard even if we are completely certain beforehand about how voters will vote (but we still need to elicit their preferences to prove that we are right) [36].

6 Strategic agents and mechanism design

A final important issue is that of strategic bidding/voting. Depending on the rules for the auction/election and the bids/votes submitted by the other agents, it may not be in an agent's best interest to truthfully report her preferences. An agent that reports preferences other than the preferences she truly has (with the aim of obtaining a better outcome for herself) is said to *misreport* or *manipulate*. If agents manipulate, this can be detrimental to the quality of the outcome: even if we manage to solve the winner determination problem to optimality with respect to the reported preferences, there is no guarantee that this results in an outcome that is good with respect to the *true* preferences, which is what we really care about. As we will see

⁴A related but distinct notion is that of *compilation complexity*, where the goal is to summarize the votes of a subset of the voters using as few bits as possible, while retaining all the information needed to compute the winner once the remaining votes become known [24, 120].

shortly, the approaches that are taken to address this problem in combinatorial auctions and in voting are generally different. This is because in combinatorial auctions, the payments can be used to remove any incentive for the bidders to misreport their valuation functions (making it optimal for each agent to report truthfully is the standard approach in *mechanism design*)—whereas in sufficiently general voting settings, any reasonable voting rule will lead to incentives to misreport in some cases.

6.1 Combinatorial auctions

To illustrate the basic idea, it is helpful to return to the single-item auction setting. We assume *quasilinear utilities*, that is, that bidder i 's utility for winning the item at price π_i is $v_i - \pi_i$, where v_i is the bidder's (true) valuation for the item; and her utility for not winning (and paying nothing) is 0. Let us first consider the first-price sealed-bid auction. It is easy to see that in this auction, it never makes sense for a bidder to bid her true valuation v_i , because if she does so, then even if she wins, her utility is $v_i - v_i = 0$. Rather, a bidder needs to bid lower than her true valuation to have any chance of obtaining positive utility.

In contrast, let us consider the second-price sealed-bid (Vickrey) auction. To see how bidder i should bid in this auction, let us first suppose that she knows all the other bids. Then, she has two choices: either bid higher than the highest other bid, b^* , and obtain utility $v_i - b^*$; or bid lower, and obtain utility 0. Clearly, she should do the former if and only if $v_i > b^*$. But this can be achieved simply by bidding truthfully (for which she does not even need to know the others' bids). Thus, it is always optimal for a bidder to reveal her true valuation in a second-price sealed-bid auction. That is, this auction is *strategy-proof*.

A natural question is whether it is possible to make combinatorial auctions strategy-proof as well by using the right payments. It turns out that this is indeed possible, by using the *Generalized Vickrey Auction (GVA)* (which is a special case of the *Clarke mechanism*, which in turn is a member of the class of *VCG mechanisms* [25, 62, 112]). The GVA works as follows: first, solve the winner determination problem, to obtain an optimal allocation S_1, \dots, S_n , with total value $V = \sum_{i=1}^n v_i(S_i)$. Now, to determine how much bidder i must pay, remove i from the auction, and solve the winner determination problem again with the remaining bidders, obtaining a total value of V_{-i} (which is at most V). Then, bidder i must pay $V_{-i} - (V - v_i(S_i))$. It is not difficult to show that the GVA is strategy-proof.

While the GVA appears to solve the problem of strategic bidding nicely, it still has a number of drawbacks. First of all, if obtaining the efficient allocation of the items is not the objective, then the GVA may not be optimal. For example, there has been significant interest in finding a combinatorial auction mechanism that maximizes expected revenue, although this turns out to be surprisingly difficult [5, 9, 38, 76, 77]. Another problem is that the GVA is very vulnerable to multiple bidders *colluding*: while there is no incentive for a single bidder to misreport by herself, a group of bidders can make themselves better off by collectively misreporting [8, 41]. A bidder can also make herself better off by pretending to be multiple distinct bidders, which is often possible in open, anonymous environments such as the Internet. That is, the GVA is not *false-name-proof*. Several combinatorial auction mechanisms have been proposed that are false-name-proof, but they come at the cost of reduced

efficiency [65, 69, 127–129]. In fact, fairly weak conditions that preclude false-name-proofness have been given [97], and a recent paper shows that the efficiency loss from using false-name-proof combinatorial auction mechanisms (that satisfy some apparently minor assumptions) is necessarily severe [69]. (An alternative approach to dealing with false-name bidding is to verify the identities of some of the bidders after the fact [28].) A final issue is that the winner determination problems must be solved to optimality: if an approximately optimal solution is used, then the resulting modified GVA is in general no longer strategy-proof. A significant body of research has focused on creating approximation algorithms for the winner determination problem that can be combined with payment schemes that make them strategy-proof [4, 47, 48, 75, 81, 86].

6.2 Voting

While the GVA mechanism solves the problem of strategic bidding in combinatorial auctions (with some significant caveats), in voting settings no such solution exists. This is fundamentally due to the lack of payments in voting settings: the Clarke mechanism, of which the GVA is a special case, can also be applied to (imaginary) voting settings in which the voters can be required to make payments. But, without payments, there is the *Gibbard-Satterthwaite impossibility theorem* [59, 108], which states that with three or more alternatives, every voting rule that is strategy-proof is either dictatorial, or is such that there is an alternative that cannot win, regardless of the votes.⁵ One may wonder if one can do better using voting rules that use randomization in their choice of the winner; however, a later paper by Gibbard [60] characterizes the class of strategy-proof randomized voting rules completely, and the result is still mostly negative in that all the rules in the class have some undesirable properties. As an aside, the class becomes even smaller if we require false-name-proofness [29], though this is mitigated if casting additional votes comes at a cost [113].

Still, something must be done to address the problem of manipulation. Many recent papers have pursued the approach of using computational hardness as a barrier to manipulation. The idea here is the following: while for any reasonable rule, there are guaranteed to be instances where there exists a beneficial manipulation (that is, a way of voting insincerely that makes the voter better off), the voter will not be able to make use of this if the manipulation is computationally too hard to discover. Indeed, there has been a number of results showing that the problem of finding a beneficial manipulation is NP-hard (for some rules, for some definitions of the manipulation problem) [11, 12, 37, 43, 52, 66, 96, 123, 126]. While this shows that there is indeed some computational barrier to manipulation, the result is not as strong as we would like. Specifically, NP-hardness is a worst-case measure of hardness: it shows that (assuming $P \neq NP$) any algorithm for finding manipulations is going to require more than polynomial time on *some* instances, but it could still be the case that the algorithm rapidly identifies a beneficial manipulation on *most* instances. It would be more satisfactory to have a result that shows that most instances are

⁵Recent work [53] considers situations where the voting language is restricted in such a way that it is not possible for voters to vote truthfully, and in this context defines a notion of voting “sincerely” that is not subject to such impossibility results.

hard to manipulate. Unfortunately, recently, various negative results have been obtained that suggest that it is usually easy to find a manipulation for reasonable rules [42, 49, 57, 94, 95, 115, 116, 118, 119, 123, 134]. Still, it seems that the book is not yet closed on using computational hardness as a barrier to manipulation. Incidentally, computational hardness has also been considered as a barrier against other types of undesirable behavior, for example by the chairperson of the election [14, 34, 54–56, 68, 96].

Another issue is that if we use preference elicitation for a voting rule that is not strategy-proof, this may introduce additional opportunities for manipulation. This is because from the queries that a voter is asked, she can infer something about how the others are voting; and she may wish to change her vote based on this information [36]. This is an issue that does not occur if we use a strategy-proof mechanism (such as the GVA): in that case, answering the queries truthfully becomes an *ex post* equilibrium. That is, it is optimal for every agent to answer the queries truthfully, as long as the others do so as well—regardless of what their preferences are.

7 Conclusions

In this article, we considered several issues that are of importance both in combinatorial auctions and in voting settings, and compared the research on these issues across the two domains.

We first considered the rules for how outcomes are chosen in combinatorial auctions and in voting. Here we noticed a major difference. Most (though not all) combinatorial auction designs agree on the objective of maximizing the efficiency of the allocation of the items, and the differences among the designs are primarily due to other factors, such as how bidders' valuations are elicited and what incentives bidders have when they bid strategically. In contrast, in voting settings, it is generally not clear what the ideal objective is, and different voting rules can be said to correspond to different objectives.

We then considered the languages in which agents express their preferences. In combinatorial auctions, there is a significant body of research on various bidding languages and their strengths and weaknesses. In voting, there is not much research on languages yet; this is presumably primarily because most work so far has dealt with settings where the number of alternatives is not extremely large, and in such settings it does not require much space to express any ranking. However, this is not true in the case of combinatorial alternative spaces, which can have exponential size. Some languages for representing preferences in combinatorial alternative spaces have been proposed, and as the interest in such settings grows, undoubtedly so will the interest in voting languages.

Subsequently, we considered the winner determination problem. In combinatorial auctions, the winner determination problem has received a lot of attention, in part because this is perhaps the first problem that needs to be solved to be able to run a combinatorial auction. In voting, when the number of alternatives is not extremely large, the winner determination problem is quite easy for most rules. But there are some rules for which it is NP-hard, and various techniques have been proposed for executing these rules nonetheless. It seems that as the focus shifts to combinatorial alternative spaces, more difficult winner determination problems will emerge.

We then considered preference elicitation, that is, sequentially querying the agents for parts of their preferences until we know enough to determine the outcome. Again, this is a problem that has received more attention in combinatorial auctions, presumably because in a general combinatorial auction, a bidder must communicate exponentially many values, so any reduction in this communication is highly desirable. Nevertheless, preference elicitation has also received some attention in voting settings. This is because even though specifying a ranking requires space polynomial in the number of alternatives, it generally requires significant deliberation effort to decide on the precise ranking, and preference elicitation can reduce the required deliberation effort. Yet again, it seems that preference elicitation will receive more attention in settings with exponentially many alternatives.

Finally, we considered strategic agents, who will misreport their preferences if this is to their benefit. Here there is a major difference between combinatorial auctions and voting. In combinatorial auctions, it is possible to remove any incentive to misreport, using techniques from mechanism design. Unfortunately, in voting settings, mechanism design provides little more than negative results, and hence research has turned to other ways to prevent misreporting—specifically, making it computationally hard to do so beneficially. Both in combinatorial auctions and in voting settings, there is growing interest in addressing new types of manipulation, such as participating in the mechanism multiple times using false identifiers. It appears that this type of manipulation is more difficult to address in combinatorial auctions than misreporting, so perhaps the techniques that will be designed to address this will be similar across the two domains. It is not clear that the approach of making manipulation computationally hard will gain traction in the combinatorial auctions literature, especially given the recent negative results about making manipulation *usually* hard in voting settings; nevertheless, there has already been some work on the hardness of various types of manipulation in combinatorial auctions [41, 106, 109].

In summary, many of the same issues are at play in combinatorial auctions and voting settings. Considering current trends in research in the two domains, it seems that the similarities will only grow. I speculate that future research on computational aspects of voting will increasingly focus on exponentially-sized alternative spaces; it seems that here in particular, considering techniques from the combinatorial auctions literature will be useful. Also, I speculate that future research on combinatorial auctions will increasingly consider settings where manipulation cannot reasonably be entirely disincentivized via mechanism design techniques; since we already face such a situation in voting settings due to results such as the Gibbard-Satterthwaite impossibility theorem, drawing connections seems helpful. Rather than developing two separate (though possibly analogous) theories for the two different domains, it would be desirable to develop a single coherent theory that can be applied to both domains. Hence, it appears that researchers in each domain would benefit from keeping a close eye on the other domain.

References

1. Ailon, N., Charikar, M., Newman, A.: Aggregating inconsistent information: ranking and clustering. In: Proceedings of the Annual Symposium on Theory of Computing (STOC), pp. 684–693 (2005)
2. Alon, N.: Ranking tournaments. *SIAM J. Discrete Math.* **20**, 137–142 (2006)

3. Apt, K., Conitzer, V., Guo, M., Markakis, E.: Welfare undominated groves mechanisms. In: *Proceedings of the Fourth Workshop on Internet and Network Economics (WINE)*, pp. 426–437. Shanghai, China (2008)
4. Archer, A., Papadimitriou, C.H., Talwar, K., Tardos, E.: An approximate truthful mechanism for combinatorial auctions with single parameter agents. In: *Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 205–214 (2003)
5. Armstrong, M.: Optimal multi-object auctions. *Rev. Econ. Stud.* **67**, 455–481 (2000)
6. Arrow, K.: *Social choice and individual values*. New Haven: Cowles Foundation, 2nd edn (1963)
7. Ausubel, L., Milgrom, P.: Ascending auctions with package bidding. In: *Frontiers of Theoretical Economics*, vol. 1, no. 1, Article 1 (2002)
8. Ausubel, L.M., Milgrom, P.: The lovely but lonely Vickrey auction. In: Cramton, P., Shoham, Y., Steinberg, R. (eds) *Combinatorial Auctions*, chapter 1, pp. 17–40. MIT Press (2006)
9. Avery, C., Hendershott, T.: Bundling and optimal auctions of multiple products. *Rev. Econ. Stud.* **67**, 483–497 (2000)
10. Bailey, M.J.: The demand revealing process: to distribute the surplus. *Public Choice* **91**, 107–126 (1997)
11. Bartholdi, J., III, Orlin, J.: Single transferable vote resists strategic voting. *Soc. Choice Welf.* **8**(4), 341–354 (1991)
12. Bartholdi, J., III, Tovey, C., Trick, M.: The computational difficulty of manipulating an election. *Soc. Choice Welf.* **6**(3), 227–241 (1989)
13. Bartholdi, J., III, Tovey, C., Trick, M.: Voting schemes for which it can be difficult to tell who won the election. *Soc. Choice Welf.* **6**, 157–165 (1989)
14. Bartholdi, J., III, Tovey, C., Trick, M.: How hard is it to control an election? Formal theories of politics, II. *Math. Comput. Model.* **16**(8–9), 27–40 (1992)
15. Betzler, N., Fellows, M.R., Guo, J., Niedermeier, R., Rosamond, F.A.: How similarity helps to efficiently compute Kemeny rankings. In: *Proceedings of the Eighth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pp. 657–664, Budapest, Hungary (2009)
16. Blum, A., Jackson, J., Sandholm, T., Zinkevich, M.: Preference elicitation and query learning. *J. Mach. Learn. Res.* **5**, 649–667 (2004)
17. Blumrosen, L., Nisan, N.: On the computational power of iterative auctions. In: *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pp. 29–43. Vancouver, BC, Canada (2005)
18. Boutilier, C., Brafman, R., Domshlak, C., Hoos, H., Poole, D.: CP-nets: a tool for representing and reasoning with conditional ceteris paribus statements. *J. Artif. Intell. Res.* **21**, 135–191 (2004)
19. Brandt, F., Fischer, F., Harrenstein, P.: The computational complexity of choice sets. In: *Theoretical Aspects of Rationality and Knowledge (TARK)*. Brussels, Belgium (2007)
20. Cavallo, R.: Optimal decision-making with minimal waste: strategyproof redistribution of VCG payments. In: *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pp. 882–889. Hakodate, Japan (2006)
21. Cerquides, J., Endriss, U., Giovannucci, A., Rodríguez-Aguilar, J.A.: Bidding languages and winner determination for mixed multi-unit combinatorial auctions. In: *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1221–1226. Hyderabad, India (2007)
22. Charbit, P., S. Thomassé, Yeo, A.: The minimum feedback arc set problem is NP-hard for tournaments. *Comb. Probab. Comput.* **16**(1), 1–4 (2007)
23. Chevaleyre, Y., Endriss, U., Estivie, S., Maudet, N.: Multiagent resource allocation with k-additive utility functions. In: *Workshop on Computer Science and Decision Theory* (2004)
24. Chevaleyre, Y., Lang, J., Maudet, N., Ravilly-Abadie, G.: Compiling the votes of a subelectorate. In: *Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 97–102. Pasadena, CA USA (2009)
25. Clarke, E.H.: Multipart pricing of public goods. *Public Choice* **11**, 17–33 (1971)
26. Cohen, W., Schapire, R., Singer, Y.: Learning to order things. *J. Artif. Intell. Res.* **10**, 213–270 (1999)
27. Conitzer, V.: Computing slater rankings using similarities among candidates. In: *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pp. 613–619. Boston, MA, USA (2006)
28. Conitzer, V.: Limited verification of identities to induce false-name-proofness. In: *Theoretical Aspects of Rationality and Knowledge (TARK)*, pp. 102–111. Brussels, Belgium (2007)

29. Conitzer, V.: Anonymity-proof voting rules. In: Proceedings of the Fourth Workshop on Internet and Network Economics (WINE), pp. 295–306. Shanghai, China (2008)
30. Conitzer, V.: Eliciting single-peaked preferences using comparison queries. *J. Artif. Intell. Res.* **35**, 161–191 (2009)
31. Conitzer, V.: Making decisions based on the preferences of multiple agents. *Commun. ACM* **53**(3), 84–94 (2010)
32. Conitzer, V., Davenport, A., Kalagnanam, J.: Improved bounds for computing Kemeny rankings. In: Proceedings of the National Conference on Artificial Intelligence (AAAI), pp. 620–626. Boston, MA, USA (2006)
33. Conitzer, V., Derryberry, J., Sandholm, T.: Combinatorial auctions with structured item graphs. In: Proceedings of the National Conference on Artificial Intelligence (AAAI), pp. 212–218. San Jose, CA, USA (2004)
34. Conitzer, V., Lang, J., Xia, L.: How hard is it to control sequential elections via the agenda? In: Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence (IJCAI), pp. 103–108. Pasadena, CA, USA (2009)
35. Conitzer, V., Rognlie, M., Xia, L.: Preference functions that score rankings and maximum likelihood estimation. In: Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence (IJCAI), pp. 109–115. Pasadena, CA, USA (2009)
36. Conitzer, V., Sandholm, T.: Vote elicitation: complexity and strategy-proofness. In: Proceedings of the National Conference on Artificial Intelligence (AAAI), pp. 392–397. Edmonton, AB, Canada (2002)
37. Conitzer, V., Sandholm, T.: Universal voting protocol tweaks to make manipulation hard. In: Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI), pp. 781–788. Acapulco, Mexico (2003)
38. Conitzer, V., Sandholm, T.: Self-interested automated mechanism design and implications for optimal combinatorial auctions. In: Proceedings of the ACM Conference on Electronic Commerce (EC), pp. 132–141. New York, NY, USA (2004)
39. Conitzer, V., Sandholm, T.: Common voting rules as maximum likelihood estimators. In: Proceedings of the 21st Annual Conference on Uncertainty in Artificial Intelligence (UAI), pp. 145–152. Edinburgh, UK (2005)
40. Conitzer, V., Sandholm, T.: Communication complexity of common voting rules. In: Proceedings of the ACM Conference on Electronic Commerce (EC), pp. 78–87. Vancouver, BC, Canada (2005)
41. Conitzer, V., Sandholm, T.: Failures of the VCG mechanism in combinatorial auctions and exchanges. In: Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS), pp. 521–528. Hakodate, Japan (2006)
42. Conitzer, V., Sandholm, T.: Nonexistence of voting rules that are usually hard to manipulate. In: Proceedings of the National Conference on Artificial Intelligence (AAAI), pp. 627–634. Boston, MA, USA (2006)
43. Conitzer, V., Sandholm, T., Lang, J.: When are elections with few candidates hard to manipulate? *J. ACM* **54**(3), Article 14, 1–33 (2007)
44. Conitzer, V., Sandholm, T., Santi, P.: Combinatorial auctions with k -wise dependent valuations. In: Proceedings of the National Conference on Artificial Intelligence (AAAI), pp. 248–254. Pittsburgh, PA, USA (2005)
45. Davenport, A., Kalagnanam, J.: A computational study of the Kemeny rule for preference aggregation. In: Proceedings of the National Conference on Artificial Intelligence (AAAI), pp. 697–702. San Jose, CA, USA (2004)
46. de Caritat, M.J.A.N. (Marquis de Condorcet): *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix*. L'Imprimerie Royale, Paris (1785)
47. Dobzinski, S., Nisan, N.: Limitations of VCG-based mechanisms. In: Proceedings of the Annual Symposium on Theory of Computing (STOC), pp. 338–344 (2007)
48. Dobzinski, S., Nisan, N., Schapira, M.: Truthful randomized mechanisms for combinatorial auctions. In: Proceedings of the Annual Symposium on Theory of Computing (STOC), pp. 644–652 (2006)
49. Dobzinski, S., Procaccia, A.D.: Frequent manipulability of elections: the case of two voters. In: Proceedings of the Fourth Workshop on Internet and Network Economics (WINE), pp. 653–664. Shanghai, China (2008)
50. Dwork, C., Kumar, R., Naor, M., Sivakumar, D.: Rank aggregation methods for the web. In: Proceedings of the 10th World Wide Web Conference, pp. 613–622 (2001)
51. Elkind, E., Faliszewski, P., Slinko, A.: Good rationalizations of voting rules. In: Proceedings of the National Conference on Artificial Intelligence (AAAI). Atlanta, GA, USA (2010)

52. Elkind, E., Lipmaa, H.: Hybrid voting protocols and hardness of manipulation. In: Annual International Symposium on Algorithms and Computation (ISAAC) (2005)
53. Endriss, U., Pini, M.S., Rossi, F., Venable, K.B.: Preference aggregation over restricted ballot languages: sincerity and strategy-proofness. In: Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence (IJCAI), pp. 122–127. Pasadena, CA, USA (2009)
54. Faliszewski, P., Hemaspaandra, E., Hemaspaandra, L.A.: The complexity of bribery in elections. In: Proceedings of the National Conference on Artificial Intelligence (AAAI). Boston, MA, USA (2006)
55. Faliszewski, P., Hemaspaandra, E., Hemaspaandra, L.A.: How hard is bribery in elections? *J. Artif. Intell. Res.* **35**, 485–532 (2009)
56. Faliszewski, P., Hemaspaandra, E., Hemaspaandra, L.A., Rothe, J.: Llull and Copeland voting computationally resist bribery and constructive control. *J. Artif. Intell. Res.* **35**, 275–341 (2009)
57. Friedgut, E., Kalai, G., Nisan, N.: Elections can be manipulated often. In: Proceedings of the Annual Symposium on Foundations of Computer Science (FOCS), pp. 243–249 (2008)
58. Fujishima, Y., Leyton-Brown, K., Shoham, Y.: Taming the computational complexity of combinatorial auctions: Optimal and approximate approaches. In: Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI), pp. 548–553. Stockholm, Sweden (1999)
59. Gibbard, A.: Manipulation of voting schemes: a general result. *Econometrica* **41**, 587–601 (1973)
60. Gibbard, A.: Manipulation of schemes that mix voting with chance. *Econometrica* **45**, 665–681 (1977)
61. Gottlob, G., Greco, G.: On the complexity of combinatorial auctions: structured item graphs and hypertree decompositions. In: Proceedings of the ACM Conference on Electronic Commerce (EC), pp. 152–161. San Diego, CA, USA (2007)
62. Groves, T.: Incentives in teams. *Econometrica* **41**, 617–631 (1973)
63. Guo, M., Conitzer, V.: Better redistribution with inefficient allocation in multi-unit auctions with unit demand. In: Proceedings of the ACM Conference on Electronic Commerce (EC), pp. 210–219. Chicago, IL, USA (2008)
64. Guo, M., Conitzer, V.: Worst-case optimal redistribution of VCG payments in multi-unit auctions. *Games Econom. Behav.* **67**(1), 69–98 (2009)
65. Guo, M., Conitzer, V.: False-name-proofness with bid withdrawal. In: Proceedings of the Ninth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS). Toronto, Canada (2010, to appear)
66. Hemaspaandra, E., Hemaspaandra, L.A.: Dichotomy for voting systems. *J. Comput. Syst. Sci.* **73**(1), 73–83 (2007)
67. Hemaspaandra, E., Hemaspaandra, L.A., Rothe, J.: Exact analysis of Dodgson elections: Lewis Carroll’s 1876 voting system is complete for parallel access to NP. *J. ACM* **44**(6), 806–825 (1997)
68. Hemaspaandra, E., Hemaspaandra, L.A., Rothe, J.: Anyone but him: the complexity of precluding an alternative. In: Proceedings of the National Conference on Artificial Intelligence (AAAI). Pittsburgh, PA, USA (2005)
69. Iwasaki, A., Conitzer, V., Omori, Y., Sakurai, Y., Todo, T., Guo, M., Yokoo, M.: Worst-case efficiency ratio in false-name-proof combinatorial auction mechanisms. In: Proceedings of the Ninth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS). Toronto, Canada (2010)
70. Konczak, K., Lang, J.: Voting procedures with incomplete preferences. In: Multidisciplinary Workshop on Advances in Preference Handling (2005)
71. Kushilevitz, E., Nisan, N.: Communication Complexity. Cambridge University Press (1997)
72. Lahaie, S., Constantin, F., Parkes, D.C.: More on the power of demand queries in combinatorial auctions: learning atomic languages and handling incentives. In: Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI), pp. 959–964. Edinburgh, Scotland, UK (2005)
73. Lahaie, S., Parkes, D.: Applying learning algorithms to preference elicitation. In: Proceedings of the ACM Conference on Electronic Commerce (EC), pp. 180–188. New York, NY, USA (2004)
74. Lang, J.: Vote and aggregation in combinatorial domains with structured preferences. In: Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI), pp. 1366–1371. Hyderabad, India (2007)
75. Lehmann, D., O’Callaghan, L.I., Shoham, Y.: Truth revelation in rapid, approximately efficient combinatorial auctions. *J. ACM* **49**(5), 577–602 (2002)

76. Likhodedov, A., Sandholm, T.: Methods for boosting revenue in combinatorial auctions. In: Proceedings of the National Conference on Artificial Intelligence (AAAI), pp. 232–237. San Jose, CA, USA (2004)
77. Likhodedov, A., Sandholm, T.: Approximating revenue-maximizing combinatorial auctions. In: Proceedings of the National Conference on Artificial Intelligence (AAAI). Pittsburgh, PA, USA (2005)
78. McGarvey, D.C.: A theorem on the construction of voting paradoxes. *Econometrica* **21**(4), 608–610 (1953)
79. Meila, M., Phadnis, K., Patterson, A., Birmes, J.: Consensus ranking under the exponential model. In: Proceedings of the 23rd Annual Conference on Uncertainty in Artificial Intelligence (UAI). Vancouver, BC, Canada (2007)
80. Moulin, H.: Almost budget-balanced VCG mechanisms to assign multiple objects. *J. Econ. Theory* **144**(1), 96–119 (2009)
81. Mu'alem, A., Nisan, N.: Truthful approximation mechanisms for restricted combinatorial auctions. *Games Econ. Behav.* **64**(2), 612–631 (2008)
82. Müller, R.: Tractable cases of the winner determination problem. In: Cramton, P., Shoham, Y., Steinberg, R. (eds) *Combinatorial Auctions*, chapter 13, pp. 319–336. MIT Press (2006)
83. Myerson, R.: Optimal auction design. *Math. Oper. Res.* **6**, 58–73 (1981)
84. Nisan, N.: Bidding and allocation in combinatorial auctions. In: Proceedings of the ACM Conference on Electronic Commerce (EC), pp. 1–12. Minneapolis, MN, USA (2000)
85. Nisan, N.: Bidding languages for combinatorial auctions. In: Cramton, P., Shoham, Y., Steinberg, R. (eds) *Combinatorial Auctions*, chapter 9, pp. 215–231. MIT Press (2006)
86. Nisan, N., Ronen, A.: Computationally feasible VCG mechanisms. *J. Artif. Intell. Res.* **29**(1), 19–47 (2007)
87. Nisan, N., Segal, I.: The communication requirements of efficient allocations and supporting prices. *J. Econ. Theory* **129**(1), 192–224 (2006)
88. Parkes, D.: Iterative combinatorial auctions. In: Cramton, P., Shoham, Y., Steinberg, R. (eds) *Combinatorial Auctions*, chapter 2, pp. 41–77. MIT Press (2006)
89. Parkes, D., Ungar, L.: Iterative combinatorial auctions: theory and practice. In: Proceedings of the National Conference on Artificial Intelligence (AAAI), pp. 74–81. Austin, TX, USA (2000)
90. Penn, M., Tennenholtz, M.: Constrained multi-object auctions and *b*-matching. *Inf. Process. Lett.* **75**(1–2), 29–34 (2000)
91. Pini, M.S., Rossi, F., Venable, K.B., Walsh, T.: Aggregating partially ordered preferences: possibility and impossibility results. In: *Theoretical Aspects of Rationality and Knowledge (TARK)*. Singapore (2005)
92. Pini, M.S., Rossi, F., Venable, K.B., Walsh, T.: Incompleteness and incomparability in preference aggregation. In: Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI). Hyderabad, India (2007)
93. Porter, R., Shoham, Y., Tennenholtz, M.: Fair imposition. *J. Econ. Theory* **118**, 209–228 (2004)
94. Procaccia, A.D., Rosenschein, J.S.: Average-case tractability of manipulation in voting via the fraction of manipulators. In: Proceedings of the Sixth International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS). Honolulu, HI, USA (2007)
95. Procaccia, A.D., Rosenschein, J.S.: Junta distributions and the average-case complexity of manipulating elections. *J. Artif. Intell. Res.* **28**, 157–181 (2007)
96. Procaccia, A.D., Rosenschein, J.S., Zohar, A.: Multi-winner elections: complexity of manipulation, control and winner-determination. In: Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI). Hyderabad, India (2007)
97. Rastegari, B., Condon, A., Leyton-Brown, K.: Revenue monotonicity in combinatorial auctions. In: Proceedings of the National Conference on Artificial Intelligence (AAAI), pp. 122–127. Vancouver, BC, Canada (2007)
98. Rossi, F., Pini, M.S., Venable, K.B., Walsh, T.: Strategic voting when aggregating partially ordered preferences. In: Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS), pp. 685–687. Hakodate, Japan (2006)
99. Rothe, J., Spakowski, H., Vogel, J.: Exact complexity of the winner problem for young elections. In: *Theory of Computing Systems*, vol. 36, no. 4, pp. 375–386. Springer-Verlag (2003)
100. Rothkopf, M., Pekeć, A., Harstad, R.: Computationally manageable combinatorial auctions. *Manage. Sci.* **44**(8), 1131–1147 (1998)
101. Russell, S., Norvig, P.: *Artificial Intelligence: A Modern Approach*, 3rd edn. Prentice Hall (2010)

102. Sandholm, T.: Algorithm for optimal winner determination in combinatorial auctions. *Artif. Intell.* **135**, 1–54 (2002)
103. Sandholm, T.: Optimal winner determination algorithms. In: Cramton, P., Shoham, Y., Steinberg, R. (eds) *Combinatorial Auctions*, chapter 14, pp. 337–368. MIT Press (2006)
104. Sandholm, T., Suri, S.: BOB: improved winner determination in combinatorial auctions and generalizations. *Artif. Intell.* **145**, 33–58 (2003)
105. Sandholm, T., Suri, S., Gilpin, A., Levine, D.: Winner determination in combinatorial auction generalizations. In: *Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)*, pp. 69–76. Bologna, Italy, (2002)
106. Sanghvi, S., Parkes, D.C.: Hard-to-Manipulate Combinatorial Auctions (2004)
107. Santi, P., Conitzer, V., Sandholm, T.: Towards a characterization of polynomial preference elicitation with value queries in combinatorial auctions. In: *Conference on Learning Theory (COLT)*, pp. 1–16. Banff, Alberta, Canada (2004)
108. Satterthwaite, M.: Strategy-proofness and Arrow's conditions: existence and correspondence theorems for voting procedures and social welfare functions. *J. Econ. Theory* **10**, 187–217 (1975)
109. Sher, I.: Optimal Shill Bidding in the VCG Mechanism (2009)
110. Tennenholtz, M.: Some tractable combinatorial auctions. In: *Proceedings of the National Conference on Artificial Intelligence (AAAI)*. Austin, TX, USA (2000)
111. Truchon, M.: Borda and the maximum likelihood approach to vote aggregation. *Math. Soc. Sci.* **55**(1), 96–102 (2008)
112. Vickrey, W.: Counterspeculation, auctions, and competitive sealed tenders. *J. Finance* **16**, 8–37 (1961)
113. Wagman, L., Conitzer, V.: Optimal false-name-proof voting rules with costly voting. In: *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pp. 190–195. Chicago, IL, USA (2008)
114. Walsh, T.: Uncertainty in preference elicitation and aggregation. In: *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pp. 3–8. Vancouver, BC, Canada (2007)
115. Walsh, T.: Manipulability of single transferable vote. In: *Proceedings of the CARE'09 International Workshop on Collaborative Agents—Research and Development*. Melbourne, Australia (2009)
116. Walsh, T.: Where are the really hard manipulation problems? The phase transition in manipulating the veto rule. In: *Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 324–329. Pasadena, CA, USA (2009)
117. Xia, L., Conitzer, V.: Determining possible and necessary winners under common voting rules given partial orders. In: *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pp. 196–201. Chicago, IL, USA (2008)
118. Xia, L., Conitzer, V.: Generalized scoring rules and the frequency of coalitional manipulability. In: *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pp. 109–118, Chicago, IL, USA (2008)
119. Xia, L., Conitzer, V.: A sufficient condition for voting rules to be frequently manipulable. In: *Proceedings of the ACM Conference on Electronic Commerce (EC)*, pp. 99–108. Chicago, IL, USA (2008)
120. Xia, L., Conitzer, V.: Compilation complexity of common voting rules. In: *Proceedings of the National Conference on Artificial Intelligence (AAAI)*. Atlanta, GA, USA (2010)
121. Xia, L., Conitzer, V., Lang, J.: Voting on multiattribute domains with cyclic preferential dependencies. In: *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pp. 202–207. Chicago, IL, USA (2008)
122. Xia, L., Conitzer, V., Lang, J.: Strategic sequential voting in multi-issue domains and multiple-election paradoxes. In: *Ninth Conference on Logic and the Foundations of Game and Decision Theory (LOFT)* (2010, to appear)
123. Xia, L., Conitzer, V., Procaccia, A.D.: A scheduling approach to coalitional manipulation. In: *Proceedings of the ACM Conference on Electronic Commerce (EC)*. Cambridge, MA, USA (2010)
124. Xia, L., Lang, J., Ying, M.: Sequential voting rules and multiple elections paradoxes. In: *Theoretical Aspects of Rationality and Knowledge (TARK)*, pp. 279–288. Brussels, Belgium (2007)
125. Xia, L., Lang, J., Ying, M.: Strongly decomposable voting rules on multiattribute domains. In: *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pp. 776–781. Vancouver, BC, Canada (2007)
126. Xia, L., Zuckerman, M., Procaccia, A.D., Conitzer, V., Rosenschein, J.: Complexity of unweighted coalitional manipulation under some common voting rules. In: *Proceedings of the*

- Twenty-First International Joint Conference on Artificial Intelligence (IJCAI), pp. 348–353. Pasadena, CA, USA (2009)
127. Yokoo, M.: The characterization of strategy/false-name proof combinatorial auction protocols: price-oriented, rationing-free protocol. In: Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence (IJCAI), pp. 733–742. Acapulco, Mexico (2003)
 128. Yokoo, M., Sakurai, Y., Matsubara, S.: Robust combinatorial auction protocol against false-name bids. *Artif. Intell.* **130**(2), 167–181 (2001)
 129. Yokoo, M., Sakurai, Y., Matsubara, S.: The effect of false-name bids in combinatorial auctions: new fraud in internet auctions. *Games Econom. Behav.* **46**(1), 174–188 (2004)
 130. Young, H.P.: Optimal voting rules. *J. Econ. Perspect.* **9**(1), 51–64 (1995)
 131. Young, H.P., Levenglick, A.: A consistent extension of Condorcet's election principle. *SIAM J. Appl. Math.* **35**(2), 285–300 (1978)
 132. Zinkevich, M., Blum, A., Sandholm, T.: On polynomial-time preference elicitation with value queries. In: Proceedings of the ACM Conference on Electronic Commerce (EC), pp. 176–185. San Diego, CA, USA (2003)
 133. Zuckerman, D.: Linear degree extractors and the inapproximability of max clique and chromatic number. *Theory of Computing* **3**(1), 103–128 (2007)
 134. Zuckerman, M., Procaccia, A.D., Rosenschein, J.S.: Algorithms for the coalitional manipulation problem. *Artif. Intell.* **173**(2), 392–412 (2009)